Remarks

Reconsideration of the above-captioned application is respectfully requested. Claims 1, 4, 13-15, 17, and 26 have been rejected under 35 U.S.C. §102 as being anticipated by Caid et al., USPN 5,619,709, Claim 5 has been rejected under 35 U.S.C. §103 as being obvious over Caid et al., and Claims 2, 3, 6-12, 18-25, and 27 have been rejected under 35 U.S.C. §103 as being obvious over Caid et al. in view of Hill, USPN 5,713,016. Claim 16 has been rejected under 35 U.S.C. §103 as being obvious over Caid et al. in view of Dumais et al., USPN 6,192,360.

Without acquiescing in any allegation not specifically addressed herein but focussing on the principal error in the rejections, the allegation that Caid et al., col. 6, lines 53-62 teach position independency (an element of each independent claim) is incorrect. The relied-upon section of Caid et al. teaches precisely the opposite:

> "For each target 202, context vectors of neighbors 203 are used to influence the context vector of target 202. The relative influence of each neighbor is weighted by two factors: 1) a function dependent on the ***neighbor's position in the window relative to the target***, and 2) a frequency function determined by the number of documents containing the neighbor stem (frequency). The closer the neighbor, and the lower the frequency function, the more "influence" the neighbor has." (emphasis mine).

Thus, Caid et al. explicitly require taking account of the position of a neighbor in a context window relative to a target in the context window, in marked contrast to, e.g., Claim 1, which requires generating a statistical evaluation of the characteristics of all of the windows such that the results are ***not*** a function of the order of the appearance of words within each window.

The examiner's contention that the last sentence in the above-quoted passage from Caid ct al. translates to position independency is thus incorrect. The relied-upon section of Caid et al. simply amplifies that the position-dependency of Caid et al. may be further refined by weighting closer words in the target's window more than distant words, but the position dependency remains. That is, the relied-upon passage of Caid et al. requires position dependency and, as an enhancement, weighting of position-dependent words depending on closeness - the opposite of what is claimed. Accordingly, since this section of Caid et al. teaches away from the present claims, and since no other section of Caid et al. appears to suggest position independency, the present claims are patentable.

Other rejections in the Office Action appear to rest in factually incorrect allegations as well. For example and without limitation, the allegation that the window 204 of Caid et al. does not contain the target 202 but only the neighbor stems 203, used to reject Claim 14 (requiring that the word around which each window is created is not included in the window) is incorrect. As clearly shown in each and every one of Figures 2A-2F, reference numeral 204 (the window reference numeral) generally points to the sequence of words correlating to reference numerals 202 and 203. Thus, when Caid et al. teaches that there are "only" three neighbors in the window of Figure 2F, it means that there are no other neighbors 203, not that the window 204 does not include the target 202, which it plainly does as shown in Figure 2F.

The Examiner is cordially invited to telephone the undersigned at (619) 338-8075 for any reason which would advance the instant application to allowance.

Respectfully submitted,

John L. Rogitz
Registration No. 33,549
Attorney of Record
750 B Street, Suite 3120
San Diego, CA 92101
Telephone: (619) 338-8075

JLR:jg

sift through all of the extraneous data to find the desired data, which may be a time-consuming process.

Another problem with conventional keyword based searches is related to the inherent properties of the human language. A keyword selected by the user may not match the words within the text or may retrieve extraneous information for a couple of reasons. First, different people will likely choose different keywords to describe the same object. For example, one person may call a particular object a [A]"bank"[@] while another person may call the same object a [A]"savings and loan"[@]. Second, the same word may have more than one distinct meaning. In particular, the same word used in different contexts or when used by different people may have different meaning. For example, the keyword [A]"bank"[@] may retrieve text about a riverbank or a savings bank when only articles about a saving bank are desirable, because the keyword does not convey information about the context of the word.

To overcome these and other problems in searching large databases considerable research has been done in the areas of Statistical Natural Language Processing, also referred to as Text Mining. This research has focused on the generation of simplified representations of documents. By simplifying document representation the ability to find desired information among a large number of documents is facilitated. One common simplification is to ignore the order of words within documents. This is often called a [A]"bag of words"[@] representation. Each document is represented as a vector consisting of the words, regardless of the order of their occurrence. However, with this approach information relating to the context and meaning of the words due to their order is lost and the ability to discriminate desired information is sometimes lost.

Other models have been developed for modeling language that do take sequences of words into account. However, such models are quite specialized and can become quite complicated. Hence they are not very useful for general text mining.

Thus, there is a need for improved techniques to assist in searching large databases. To this end there is also a need for improvements in Statistical Natural Language Processing that overcomes the disadvantages of both the models that take the sequences of words into account and those that do no take the sequence of words into account.

The present invention has carefully considered the above problems and has provided the solution set forth herein.

SUMMARY OF THE INVENTION

A computer-implemented system and method is disclosed for retrieving documents using context-dependant probabilistic modeling of words and documents. The present invention uses multiple overlapping vectors to represent each document. Each vector is centered on each of the words in the document, and consists of the local environment, i.e., the word[ ]s that occur close to this word. The vectors are used to build probability models that are used for predictions. In one aspect of the invention a method of context-dependant probabilistic modeling of documents is provided wherein the text of one or more documents are input into the system, wherein each document includes human readable words. Context windows are then created around each word in each document. A statistical evaluation of the characteristics of each window is then generated, where the results of the statistical evaluation are not a function of the order of the appearance of words within each window. The statistical evaluation includes the counting of the occurrences of

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

1. Introduction

Referring initially to Figure 1, a context-dependant probabilistic document modeling system is shown, generally designated 10, for storing and retrieving documents. As shown, the system 10 can include a computer 12 including a respective input device 14 such as a keyboard with, e.g., a point and click device, and an output device 16, such as a monitor, printer, other computer, or computer network. Also, the computer 12 accesses a database 18, which contains a large amount of textual data that a user desires to access. In particular, a user will input a query 2[2]0 for the computer 12 to find a particular kind of information in the database 18. In response, the computer 12, using a context-dependant probabilistic modeling module 22, will find the desired data and provide a response 24 to the user.

The computer 12 can be a personal computer made by International Business Machines Corporation (IBM) of Armonk, N.Y. Other digital processors, however, may be used, such as a laptop computer, mainframe computer, palmtop computer, personal assistant, or any other suitable processing apparatus. Likewise, other input devices, including keypads, trackballs, and voice recognition devices can be used, as can other output devices, such as data storage devices.

In any case, the processor of the computer 12 accesses the context-dependant probabilistic document modeling module 22 to undertake the logic of the present invention, which may be executed by a processor as a series of computer-executable instructions. The instructions may be contained on a data storage device with a computer readable medium, such as a computer diskette 26 shown in Figure 2 having a computer usable medium 28 with code elements. Or, the instructions may be stored on random access memory (RAM) of the computer 12, on a DASD array, or on

The probabilistic models used with the present invention are conditional probabilities where the condition is given by words, or windows. Bayes[=]'s Rule is described by equations (1) and (2) in Figure 3, where d is the variable to be modeled and $O = o_1,Y,o_M$ is the environment. p(d) is the prior probability for variable d. Users[=]' preferences can be encoded in this distribution, e.g., so that document or terms deemed more interesting will be favored as determined by prior usage of a retrieval system.

If a text (e.g., a document itself or a sentence) is used for input, context windows are created for the text. Each window is evaluated and the results are combined, preferably by averaging the probability assessments, but other combinations are possible. Since the result is normalized, input length is taken into account. If the input context is smaller than the window size the models can be used directly since the result is equivalent.

2.3 Models for Documents, Categories and Words

The models used in the present invention can be used to predict three things:

1. Document membership. The model is p(d¦O). The predicted variable d may or may not be appended with the specific context window number.

2. Document Category, p(t¦O). The specific context window may also be included here.

3. Word in the center of the environment. Model: p(c¦O),

where O is the context window. These models will be examined individually.

The Document Model uses a document identifier that is modeled from the context window, p(d¦O). There are two uses for this model:

1. By evaluating p(d¦O) with any context and finding the document I.D., d, for which this quantity is maximized, this model can be used for document retrieval.

in Figure 3. It is usually the case that $p(o_1,Y,o_M)$ is fixed when evaluating $p(d¦o_1,Y,o_M)$ over all d. It then becomes a normalizing factor since $Ó^N{}_{i=1}$ $p(d_i¦o_1,Y,o_M) = 1$. See equation (5) in Figure 3.

To use this model is necessary to remember all p(d) and all $p(o_i¦d)$. Since $p(o_i¦d)$ is defined as $p(o_i,d)/p(d)$ it is necessary to keep track of all pair-wise probabilities $p(o_i,d)$. The probabilities are estimated by counters, as described below. For computational reasons it is often useful to write this in logarithmic form, as shown in equation (6) in Figure 3.

3.3 Mutual Information Simple Bayes

An alternate representation of Simple Bayes is sometimes used. Assume, in addition to equation (3), that also equation (7), shown in Figure 3, is valid. The conditional probability can then be written as equation (8). $p(o_i¦d)$ is then the same as $p(o_i,d)/p(o_i)p(d)$. Taking logarithms this is called Mutual Information, or sometimes Point-wise Mutual Information. It is defined between variables s and y as shown in equation (9) in Figure 3. Defining $B_d = \log_2 p(d)$ it is possible to rewrite the logarithm of equation (2) as equation (10) shown in Figure 3.

The conditional probability can thus be modeled as a sum of the pair-wise mutual information values. The B terms are bias values that are modified by the pair-wise correlations, as measured by Mutual Information. Mutual Information has been used for correlations such as word sense disambiguation.

Since it is known that [A]"uninteresting"[@] combinations have values close to one, this fact can be used for pruning down the number of combinations that need to be stored. The most uninteresting combinations will be for common words such as [A]"the"[@], etc. The B-term is a [A]"bias"[@] value that indicates how common a word is in the entire collection, the prior probability.

3.4 Pruning

The two Simple Bayes models both work by adding values to a bias. Some of the added values are small and can be removed or pruned from the model. A threshold is selected and all values below that threshold are removed for the standard case, or all pairs with an absolute value of the mutual information or logarithm of the conditional probability below a threshold are removed.

It has been found by using the present invention on actual databases that the actual number of useful pairs can be as low as 1/1000 of the possible pair combinations for center word prediction. A collection of 5,000_documents had approximately 39,000 unique words and about 1,000,000 pairs after mild pruning (at threshold 8), a reduction of 30% compared to keeping all combinations. This is a large number but quite manageable. The growth of the number of pairs is also largest in the beginning since local vocabularies are limited.

In general, it should only be necessary to prune the word prediction model since the other models do not grow to the same sizes. The pruning is done periodically as the pair-wise counters grow in numbers. It is possible to prune by monitoring how much memory is used. Since the pruning is done periodically, the number of pairs will go up and down. Some pairs that have disappeared can reappear at a later stage if their use is increased in later seen documents.

3.5 Probability Estimation

The Simple Bayes probability estimates are found through counts in accordance with the invention. Let c_i be the number of times word i occurs and c_{ij} be the number of times the pair of i and j occur. There are N words in total. Then the relevant probabilities are as described in equations (11), (12), (13), and (14) in Figure 3.

Learning. Mixture Models are more expensive to learn compared to Simple Bayes but are made manageable by using the Expectation-Maximization (EM) algorithm. It is also possible to build the models using only a subset of the training data.

Mixture Models are a type of generative model where the data is assumed generated by a model. The parameters for the model are then chosen so that the likelihood of the data given the model is maximized. This is called Maximum Likelihood Estimation.

Similar vectors are grouped together to form clusters or mixtures. The clusters define probability distributions that are linearly combined. The clusters work as generators and it is assumed that each data point can be generated by a unique mixture. Mixture Models can be viewed as a [A]"soft"[@] form of classification, or a soft form of clustering, where each data point is allowed to be allocated to several clusters instead of just one. Each point thus has a probability distribution over the clusters. This allows for flexible and accurate models.

4. Implementations

Figure 4 is a flow chart of a process for the context-dependant probabilistic modeling of words and documents in accordance with one embodiment of the invention. A text is first input into the document retrieving system 10 of the invention, as shown at block 30 in Figure 4. A set of windows is generated around each word in the document, as indicated at block 32. A statistical evaluation of all the windows and documents is then performed, as shown in block 34. This will include collecting statistical counts of each element in the windows as well the each pair-wise counts, in the example shown in Figure 5-7 and described below. The order of the words within each window is not considered, only the words themselves and the counts of the numbers of each word present. The center word within each window is not contained in the window and the window may be symmetric or asymmetric in size around the center word.

The results are then combined, as shown in block 36. An appropriate statistical model, such as Simple Bayes, is then generated and applied to the combined results, as shown blocks 38 and 40. The final results are then calculated based on the counts, as indicated in block 42. For example, the results may be statistical information that is used to retrieve a document, extract features from a document or find the center word in a window.

A specific example of the use of the context-dependant probabilistic modeling techniques of the present invention is illustrated in Figures 5-7. Figure 5 shows an example of two documents, Document 1 and Document 2, which each contain five words. The task of the model is to associate the windows with the documents. This model is useful for document retrieval and feature extraction. In this case the specific context window membership is not modeled to simplify the presentation.

Overlapping windows are created around each word in the documents. In this example, the window size is minus 2 words to plus 2 words around each word. The windows are shown in Figure 6. Statistics about singular word occurrences and pairs are then collected. In particular, for each window and variable (in this case the variable is the document number): 1) statistics are collected for each variable; and 2) pair-wise statistics are collected for variables and each element in the window.

Figure 7 shows the statistics collected for each variable. For the first variable, document 1, the single counter finds 5 words in the document. Likewise, for the second document the single counter finds 5 words in the document. For the word [A]"the"[@], the counters find that the word appears seven times in the windows. Likewise, the word [A]"quick"[@] appears 3 times in the windows. This counting process is repeated for each other word.

As shown in Figure 7 the pair-wise counter finds that the pair [A]"1 - the"[@] appears three times. In other words, the word [A]"the"[@] appears three times in the document 1 windows. Likewise, the word [A]"quick"[@] appears three times in the document 1 windows. This process is repeated for each pair-wise combination of words within the windows and document numbers.

Using the results from the counters shown in Figures 7 and 8, the probabilities for any document can be estimated given a number of words, i.e. p(d¦w_1,Yw_n). The equations for this are given in Figure 3. In particular, probabilities are estimated by dividing by N,p(x) = C(x)/N, where C(x) is the number of times x occurs. For example, p(fox) = 3/28 = 0.1071. Better estimates for probability are possible using the equations in Figure 3. In this case, p(fox) would be estimated by (3+1)/(28+11) = 0.1026.

Conditional probabilities p(x¦y) are estimated by C(x,y)/C(x). Hence, p(brown¦1) is C(1,brown)/C(1). For example, p(1¦brown) is the probability of seeing document 1 if the word seen is brown. Thus p(1) + p(brown¦1)/p(brown) = 5/28 + 3/5/3 = 0.38. Similarly, for document 2: p(2¦brown) = 5/28 + 0/3/5 = 0.18. Since this model is an approximation, the values don[=]"t sum to 1. Normalization is done so that p(1¦brown) + p(2¦brown) = 1. Hence, it is more likely that the document is 1 than 2 if the word is brown.

In order to speed up retrieval of documents and related words in the system a specific database format can be used. To find the conditional probabilities of a word that is related to some other words, e.g., a query, the words that are related to the query words must be known using the